DOCUMENT RESUME

ED 249 933 IR 011 162

AUTHOR Sheingold, Karen; And Others
TITLE Classroom Software for the Information Age. Technical Report No. 23.
INSTITUTION Bank Street Coll. of Education, New York, NY. Center for Children and Technology.
SPONS AGENCY Carnegie Corp. of New York, N.Y.
PUB DATE [83]
NOTE 12p.; For related document, see IR 011 340.
PUB TYPE Viewpoints (120)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Graphics; *Computer Software; Cost Effectiveness; *Databases; Data Processing; Editing; *Evaluation Criteria; Information Retrieval; Information Storage; *Media Selection; *Word Processing
IDENTIFIERS *Database Management Systems

ABSTRACT

Consumers and producers of educational software must make decisions about what kind of software to buy and create. Both groups must base these decisions on criteria that consider what it is important to learn in our technological era, what is workable, and what is currently practical and cost-effective. Five criteria that are central to these decisions are met by "tool" software, or software that helps users access, organize, and reorganize information of a particular type, as do word processors, database management systems, music editing systems, and graphic editors. Tools do not specify what users must do but define the activities in which they can engage. Tool software meet these important criteria because they (1) help students acquire and use strategies for managing information; (2) help students learn about important aspects of technology; (3) can be assimilated into existing classroom curricula while providing the possibilities of new classroom activities and learning; (3) are cost-effective for both developers and purchasers; and (5) can be developed for the personal computers currently in use in schools. Creating software that benefits students and teachers will require collaborative efforts among technology specialists, educational practitioners, and researchers. Ten references are listed. (LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

CENTER FOR CHILDREN AND TECHNOLOGY

ED249933

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Denise Newman
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Classroom Software for the Information Age

Karen Sheingold, Jan Hawkins, and
D. Midian Kurland

Technical Report No. 23

IR011362

Bank Street College of Education 610 West 112th Street NY, NY 10025

2

Classroom Software for the Information Age

Karen Sheingold, Jan Hawkins, and
D. Midian Kurland

Technical Report No. 23

IR011362

CENTER FOR CHILDREN AND TECHNOLOGY
Bank Street College of Education
610 West 112th Street
New York, NY 10025

CLASSROOM SOFTWARE FOR THE INFORMATION AGE*

Karen Sheingold, Jan Hawkins, and D. Midian Kurland

There is no question that, from now on, computer technology will be part of the formal and informal education of children. The critical questions are how technology will contribute to education, and what the outcomes of such a contribution are likely to be. Technology can be used to do some of the same things that are now being done in classrooms, or can facilitate new kinds of activities and, it is hoped, new kinds of learning. While computer hardware is the most visible part of the "technological revolution," the future of microcomputer technology in classrooms hinges on appropriately designed software and on how teachers are able to incorporate it into their classrooms.

Decisions are now being made by schools about what kind of software to buy, and by publishers and developers about what kind of software to produce. For both groups--consumers and producers of educational software--there are important criteria to apply in advance of such decisions. Considerations about what it is important to learn in our technological era and what is workable in classrooms, as well as what is currently practical and cost-effective, all come into play. Five criteria are central:

1. The information age is making possible new kinds of activities and learning in classrooms, and is also making changes in what students need to know. The vast amounts of information being made available via computer, cable, satellite, and fiberoptic technologies will increasingly challenge human capacities to deal with information. While students will, in principle, have access to richer sources of knowledge and information than ever before, there is the real possibility that they will be overwhelmed by the quantity (and variable quality) of what is available. Successful students will be those who can effectively organize, access, manipulate, evaluate, query, and communicate about information. Students will need to apply these skills in many curricular domains--science, history, social studies, and literature, to name just a few. Plans for using and developing software, therefore, should be responsive to the need for increased emphasis on information-management skills.

---

*The preparation of this paper was made possible by a grant from the Carnegie Corporation of New York.

2. Some level of comfort with and mastery of computer-based technologies is viewed as a "new basic" in our schools. The information tools which will figure significantly in the adult work life of today's students can become part of the life of the classroom. Software which enhances students' ability to function in a technological society deserves priority over that which does not.

3. Software must be able to fit in with the existing organization of classrooms and curricula while, at the same time, providing the possibility of new activities and new learning. Teachers must be able to recognize software as something that can help them meet curricular goals. Yet, given the rich potential of the new classroom technology, the software should be usable for new learning and new purposes as these are designed and discovered by students and teachers.

4. Software must be cost-effective to producers and consumers alike. The economic realities of the educational software market are such that neither producers nor consumers have adequate resources for development or purchase. Relatively inexpensive software that has broad applications for the consumer should take priority over more costly or limited applications.

5. Effective software for classroom use must be developed for the hardware which schools now have available or will have within the next two years. Although hardware development is proceeding at breakneck speed, there is every reason to believe that schools will not be able to keep up. Therefore, new educational software for school use must be compatible with current microcomputer capabilities.

The type of software which meets these criteria is that which we refer to simply as "tools." Software tools have great potential for incorporation into learning contexts, but have not yet been well explored for educational purposes. Tools are those pieces of software which help users access, organize, manipulate, and communicate about information. Most kinds of tool software enable people to enter, revise, organize, and reorganize information of a particular type: text, in the case of word processors; facts and figures, in the case of database management systems; musical notes and rhythms, in the case of music editing systems; and charts, graphs, and designs, in the case of graphics editors. These are referred to as tools because, like pencils, paper, rulers, typewriters, and calculators, they help people to accomplish tasks but do not specify the content of these tasks.

Tool software can be contrasted with other kinds of software on many dimensions, perhaps the most important being the extent to which the program specifies and constrains what the user does. In games and

computer-assisted instruction, the content of the program is completely specified and the user's role is to respond to it. The interaction is "software-driven," with the user's goal to provide correct answers or make moves that score points. Programming languages are on the opposite end of the continuum, being most unlike computer-assisted instruction and least specific with respect to what the user may do. While different programming languages are more and less appropriate for certain purposes, each can be used by the expert programmer to accomplish a wide variety of goals.

Tool software, then, fits somewhere in between programming languages and computer-assisted instruction. It does not specify what the users must do, but rather defines broadly the type of activity they can engage in (e.g., writing, musical composing, or fiscal modeling). Tools are widely used in business, universities, and industry. Yet, with a few exceptions, they have not been designed appropriately for the precollege population.

Word processors and database management systems are examples of what tool software does and how it can be used in the classroom. Word processors are tools for writing and editing; students can use them to compose essays, poems, letters, stories, or newspaper articles. Students can easily correct what they write, insert new text at any point, delete words, phrases, sentences, and paragraphs, and move text around. All of this can happen before they print out their work, so that their printed copy is clean, neat, and "finished." Thus students can, at least in principle, come to see text as flexible and writing as a multi-staged process.

Database management systems (DBMS) are programs that allow users to create multiple files--not unlike traditional note cards--containing information on any topic. Unlike note cards, however, the files once entered into the data base can be instantly sorted, reorganized, edited, updated, printed out, or turned into summary reports. DBMS are extremely flexible organizers of information which can be used for note-taking, outlining, and other prewriting activities. They also make large amounts of information easily accessible to users who want to prepare reports or simply to learn about some content area.

For example, a class studying sea animals could decide on a useful file structure for building a sea animal data base (e.g., enter the animal's name, size, eating habits, color, habitat, and a paragraph highlighting the interesting features of the animal). After researching different animals, students could add them to the classroom's data base, which could then be used by the whole class to answer such questions as the relationship between the animals' size, habitat, and what they eat. In addition to helping students learn about sea

animals, such a project could present opportunities for learning the value of good data organization, the usefulness of different retrieval strategies, and problems with maintaining quality and accuracy in an open-access data system. The teacher would necessarily play a critical role in facilitating such learning.

Tools and Children's Learning

Tool software, unlike any other type available for education, meets the five criteria listed above. It provides students with the opportunity to access, organize, and manipulate information in new ways. Thus tools may offer students practice with certain cognitive strategies for making use of information. These strategies may be important for effective learning and use of knowledge.

One of the most highly valued accomplishments of the early and middle school years is students' learning to understand and work with information from various sources, particularly texts; that is, students must develop principled, organized, and coherent strategies for reading, comprehending, and extracting appropriate information from books and other written materials (Brown, Bransford, Ferrara & Campione, 1983). In order to understand and to improve students' information-handling skills, researchers have analyzed how successful and unsuccessful students go about studying and learning from texts. For example, Brown and Smiley (1977) report that successful students are able to identify and abstract the important material. In addition, these students develop a variety of strategies for reorganizing and summarizing information in order to learn actively. They learn to delete redundancy, to select a topic, and to identify superordinate themes and events (Brown, 1980; Brown & Day, 1983; Brown, Smiley & Lawton, 1978). They are able to clarify the significance of information in terms of their own knowledge by looking for the relevance of facts in a text and by examining the relationships between these facts and what they already know (Bransford, Stein, Shelton & Owings, 1981).

Thus, research findings indicate that successful learning is a function of active strategic engagement with the material in order to make sense of it in terms of what is already known. Active strategic engagement involves identification of important themes, topicalization, abstraction, and reorganization of material. A student's ability to manipulate information in these ways contributes to her understanding of the material in the text and to the expansion of her knowledge. Successful students learn that studying is a "purposive, attention-directing, self-questioning act" (Brown, Bransford, Ferrara & Campione, 1983).

Software tools make explicit some of these important learning strategies. Effective use of a DBMS to organize, store, and retrieve information entails topicalization, abstraction, and summarization skills. It is reasonable to hypothesize that working with this software tool offers students practice with important basic skills for effective learning and use of information.

Tools make possible the flexible manipulation of information that the drill-and-practice software most typically used in classrooms does not. The latter delivers small amounts of highly organized information (lessons or tests) such that the students' only choice is to indicate the answer to a question or problem. If students answer correctly, they are rewarded and then given similar or more difficult problems. If incorrect, they are told so and are cycled through problems like the one they missed. While, under some circumstances, such software may improve children's factual learning, it does not appear to help them acquire and use strategies for managing information (Becker, 1982; Ragosta, Holland & Jamison, 1982).

Tools for Learning About Technology

Tools also meet our second criterion, that software should help children to learn about important aspects of technology. Tools are examples of software used by society at large for other than educational purposes. Students will need to use information tools effectively and comfortably in both the home and workplace as adults. A recent report from the Office of Technology Assessment (1982) states: "The information revolution places new demands on individuals, changing what they must know and what skills they must have to participate fully in modern society."

Is there other software that could effectively help students learn about technology and how to use it? Programming languages are an obvious candidate, and programming is already widely taught. There are serious debates in the educational community about the centrality of programming to technological literacy, and programming itself is undergoing profound changes. In the near future, programming will be done through highly specialized, domain-specific languages, rather than with the complex, general-purpose languages in use today. If programming is taught for the purpose of technological literacy, then it is important to consider the abstract computational concepts embodied in programming and to anticipate what programming will actually involve in the near future. However debates about technological literacy are resolved, there is no doubt that the use of powerful software tools will be seen as central to such literacy.

Tools and the Curriculum

Tools meet our third criterion, since they can be assimilated into classroom curricula as they now exist, while providing the possibilities of new classroom activities and learning. Since they are tools, they can be used to enhance learning tasks rather than to deliver new content or suggest new curricular areas. Teachers can understand what they are good for and how to make use of them, provided the software is well designed for the educational context. For example, a word processor easily fits into a teacher's conception of writing tasks. Once teachers use such tools for their regular curriculum, the tools themselves can open up new kinds of activities in a classroom. Students can use a word processor for typical classroom writing activities such as composing an essay. It can also make possible new activities, such as group writing projects in which students jointly compose and successively edit one piece of work. Students can undertake large projects with greater ease, such as creating magazines and newspapers, along with the extensive writing and revising such work entails.

The other type of software which fits into the classroom easily is drill and practice. It is recognizable by teachers as something they normally use and, like the familiar workbooks and dittoed sheets, is easily assimilable. However, teachers often find that, because they are so specific, such programs don't match their curricular requirements. At the other extreme is programming which, for a regular classroom teacher (not a teacher of a "programming course"), is often a perplexing activity precisely because its connection with the ongoing curriculum is difficult to make. Tools occupy that important middle ground of being recognizably relevant, yet offering the potential of going beyond what usually happens in the classroom.

Tools as Cost-Effective for Developer and Consumer

Tools also meet our fourth criterion, that software be cost-effective for both developers and purchasers. While tools are (at great expense) constantly being refined, developed, and made more sophisticated for business purposes, tools for classroom use are not likely to require long design time and high development costs. Since there are already many examples of tools on the market, it may be possible to adapt such tools for education rather than to create something completely new. Computer-assisted instructional packages, on the other hand, are expensive to develop. To create a comprehensive set of drill-and-practice or tutorial programs to cover one subject area requires a number of different programs and can cost several million dollars.

Perhaps more important than the cost to the developer is the cost to the consumer. When a school buys a tool, it is buying something that can be used in different classrooms for varied purposes over a long period of time. A database system can be used just as well for sixth grade social studies as for eleventh grade science. While tools generally cost more than individual games or drill-and-practice programs, they are cost-effective because they are more adaptable and, as a consequence, schools need fewer of them.

Tools Within Current Hardware Capabilities

Tool software can be developed for the personal computers currently in use in schools. While it is always the case that additional memory and processing capacity makes the computer more powerful and flexible, there is nothing about the design of tool software per se which requires more sophisticated machines than those now used in classrooms. However, significantly more powerful machines will be needed to run interactive systems developed as a result of current work in artificial intelligence.

While scientists in artificial intelligence are creating intelligent computer-assisted instruction (ICAI) programs with significant educational potential, the development and hardware costs for such software make it prohibitive in the near term. The current costs for computer systems capable of running ICAI packages range from $10,000 to the hundreds of thousands. The prospects are slim that schools will have such machines available by the 1990s, if then. Such an optimistic date will depend on significant corporate or federal support for ICAI for educational purposes. Thus far, interest has been limited to ICAI to train technicians and professionals in medicine, industry, and the military, rather than to teach school children.

What Needs to Be Done

We have identified a type of software--tools--which holds important promise for children's learning and for use in classrooms, and which can now be developed at a reasonable cost. However, this is only a beginning. Creating software that benefits students and teachers in classrooms will require collaborative efforts among technology specialists, educational practitioners, and researchers. All must contribute to the design of the software, and the software itself must be revised and refined in the light of what researchers find out from students and teachers. Once the design, testing, revision, and extensive classroom use has resulted in a viable educational product, it will be possible to study the intellectual impact of such software on students' learning.

If technology is to make a significant contribution to education, we must make thoughtful choices about how it can best be used. Having made such choices, educators, researchers, and developers must collaborate in the design and production of effective software. Such collaborative efforts will require a significant investment of resources, without which there is little reason to hope that new technology will make a difference to the educational life of the classroom.

References

Brown, A. L., Bransford, J. D., Ferrara, R. A., & Campione, J. C. Learning, remembering, and understanding. In J. Flavell & E. Markman (Eds.), Handbook of Child Psychology (Vol 3). New York: John Wiley & Sons, 1983.

Brown, A. L., & Smiley, S. S. The structural units of prose passages: A problem of metacognitive development. Child Development, 1977, 48, 1-8.

Brown, A. L. Metacognitive development and reading. In R. J. Spiro, B. Bruce, & W. Brever (Eds.), Theoretical issues in reading comprehension. Hillsdale, NJ: Erlbaum, 1980.

Brown, A. L., & Day, J. D. Macrorules for summarizing texts: The development of expertise. Journal of Verbal Learning and Verbal Behavior, 1983, 22, 1-14.

Brown, A. L., Smiley, S. S., & Lawton, S. Q. C. The effects of experience on the selection of suitable retrieval cues for studying texts. Child Development, 1978, 49, 829-835.

Bransford, J. D., Stein, B. S., Shelton, T. S., & Owings, R. A. Cognition and adaptation: The importance of learning to learn. In M. J. Harvey (Ed.), Cognition, social Behavior and the environment. Hillsdale, NJ: Erlbaum, 1981.

Brown, A. L., Bransford, J. D., Ferrara, R. A., & Campione, J. C. Learning, remembering, and understanding. In J. Flavell & E. Markman (Eds.), Handbook of Child Psychology (Vol 3). New York: John Wiley & Sons, 1983.

Becker, H. J. Microcomputers in the classroom--dreams and realities (Report No. 319). Center for Social Organization of Schools, The Johns Hopkins University, Baltimore, MD, 1982.

Ragosta, M., Holland, P. W., & Jamison, D. T. Computer-assisted instruction and compensatory education: The ETS/LAUSD study. The executive study and policy implications. Princeton, NJ: Educational Testing Service, June 1982.

Office of Technology Assessment. Information technology and its impact on American education (OTA-CIT-188). Washington, DC: Author, 1982.